CrossLink Programming and Configuration Usage Guide

Technical Note

FPGA-TN-02014-1.5

July 2021

Disclaimers

Lattice makes no warranty, representation, or guarantee regarding the accuracy of information contained in this document or the suitability of its products for any particular purpose. All information herein is provided AS IS and with all faults, and all risk associated with such information is entirely with Buyer. Buyer shall not rely on any data and performance specifications or parameters provided herein. Products sold by Lattice have been subject to limited testing and it is the Buyer's responsibility to independently determine the suitability of any products and to test and verify the same. No Lattice products should be used in conjunction with mission- or safety-critical or any other application in which the failure of Lattice's product could create a situation where personal injury, death, severe property or environmental damage may occur. The information provided in this document is proprietary to Lattice Semiconductor, and Lattice reserves the right to make any changes to the information in this document or to any products at any time without notice.

Contents

Figures

Tables

Acronyms in This Document

A list of acronyms used in this document.

Acronym	Definition
CRC	Cyclic Redundancy Check
EBR	Embedded Block RAM
GOE	Global Output Enable
GSR	Global Set Reset
GWDIS	Global Write Disable
I^2C	Inter-Integrated Circuit
LUT	Look Up Table
MSPI	Master Serial Peripheral Interface
NVCM	Non-Volatile Configuration Memory
POR	Power On Reset
SDM	Self-Download Mode
SSPI	Slave Serial Peripheral Interface

1. Overview

The Lattice Semiconductor CrossLink™ is an SRAM-based Programmable Logic device that includes an internal Non-Volatile Configuration Memory (NVCM), as well as flexible SPI and I^2C configuration modes. The CrossLink device provides a rich set of features for the programming and configuration of the FPGA. Many options are available for building the programming solution that fits your needs. This document describes each of the options in detail.

2. CrossLink Features

The key programming and configuration features of the CrossLink devices include:

- Instant-On configuration from internal NVCM – powers up in milliseconds
- Single-chip, secure solution
- Multiple programming and configuration interfaces
  - Self-Download
  - Slave SPI (SSPI)
  - Master SPI (MSPI)
  - Dual Boot
  - I^2C
- Optional dual boot with external SPI memory
- Optional security bits for design protection

3. Definition of Terms

This document uses the following terms to describe common functions:

- BIT – The BIT file is the configuration data for the CrossLink device that is stored in an external SPI Flash. It is a binary file and is programmed unmodified into the SPI Flash.
- Configuration – Configuration refers to a change in the state of the CrossLink SRAM memory cells.
- Configuration Data – This is the data read from the non-volatile memory and loaded into the FPGA's SRAM configuration memory. This is also referred to as a bitstream, or device bitstream.
- Configuration Mode – The configuration mode defines the method the CrossLink device uses to acquire the configuration data from the non-volatile memory.
- Dummy Byte – A dummy byte is any data in which the numeric value is considered to be *don't care*. In some cases, external devices controlling the resident FPGA scan in dummy bytes as a requirement of the protocol.
- Feature Row HW Default Mode – Feature Row Hardware Default Mode is a blank Feature Row. All the Feature Row bits equal to zero.
- Number Formats – The following nomenclature is used to denote the radix of numbers
  - 0x – N umbers preceded by '0x' are hexadecimal
  - b (suffix) – Numbers suffixed with 'b' are binary
  - All other numbers are decimal
- NVCM – Internal, lowest-cost, secure, one-time programmable Nonvolatile Configuration Memory. The BIT file can be programmed directly into the internal NVCM. You do not need to know where an actual page of the configuration data starts. The CrossLink device configuration engine handles the parsing in the NVCM to SRAM transfer.
- Port – A port refers to the physical connection used to perform programming and some configuration operations. Ports on the CrossLink device include SPI and I^2C physical connections.
- Programming – Programming refers to the process used to alter the contents of the internal or external non-volatile configuration memory.
- User Mode – The CrossLink device is in User Mode after configuration and the wake-up sequence have completed.

4. Configuration Process and Flow

Before it is operational, the FPGA goes through a sequence of states, including initialization, configuration and wake-up. Figure 4.1 shows the configuration flow.

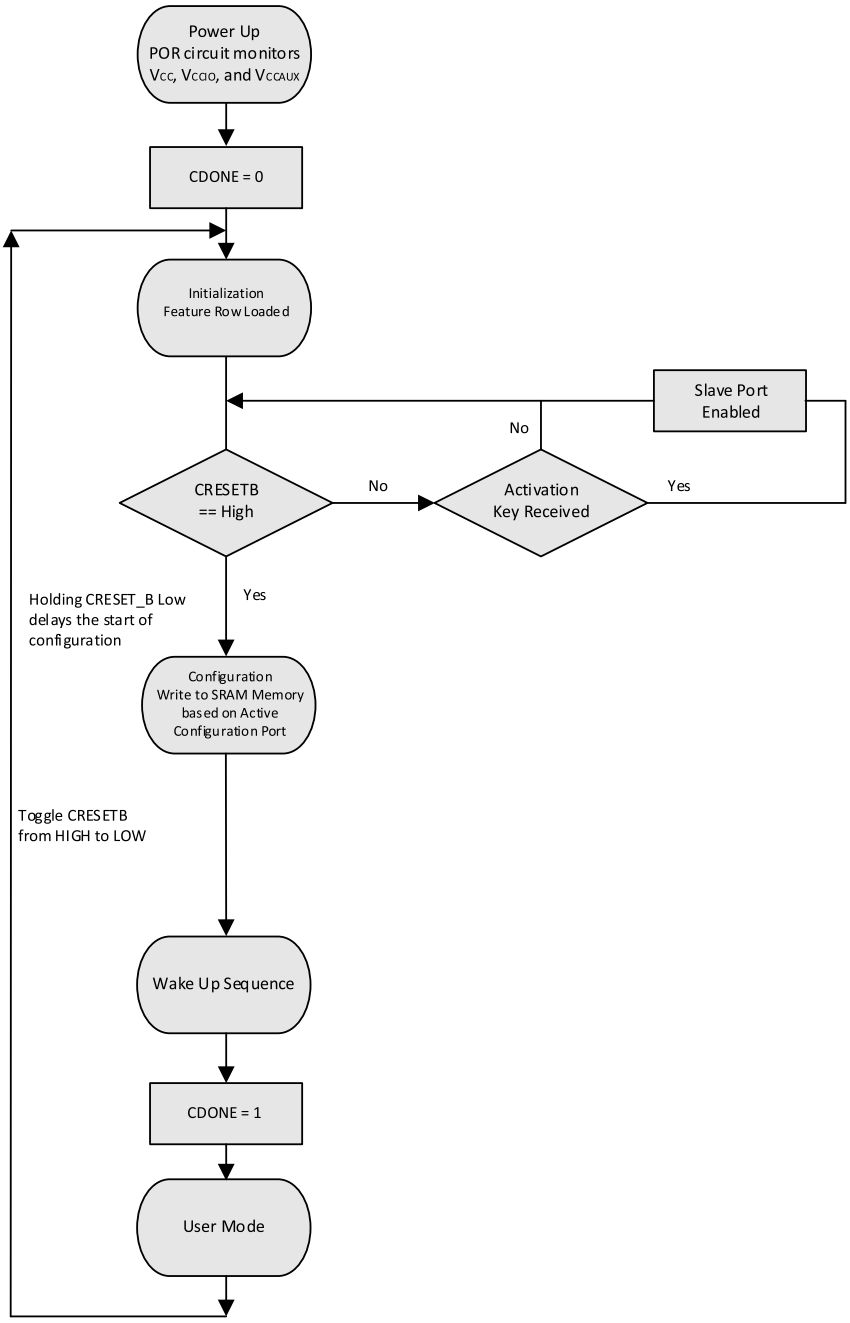

Figure 4.1. Configuration Flow

The CrossLink sysCONFIG ports provide industry standard communication protocols for programming and configuring the FPGA. Each protocol provides a way to access the CrossLink device's internal NVCM, or to load its configuration SRAM.

The sysCONFIG ports capable of accessing the NVCM have a priority order. The MSPI configuration port does not have the ability to alter the NVCM space, and as a result is not a factor in the sysCONFIG port priority scheme.

4.1. Power-up Sequence

Power must be applied to the CrossLink device for it to operate. For a short period of time, as the voltages applied to the system rise, the I/O may drive unexpected logic values until the device completes the Power On Reset (POR). All external devices should allow the FPGA to start driving valid logic levels before sampling their values.

As power continues to ramp, a Power On Reset circuit inside the FPGA becomes active. The POR circuit, once active, ensures the external I/O pins are in a high-impedance state. It also monitors the V_{CC}, V_{CCIO0} and V_{CCAUX} input rails. Refer to CrossLink Family Data Sheet (FPGA-DS-02007) for exact Power On Voltage levels and power sequencing requirements.

When POR conditions are met, the POR circuit releases an internal reset strobe, allowing the device to begin its initialization process. Once completed, the CrossLink device drives CDONE LOW.

4.2. Initialization

The CrossLink device enters the memory initialization phase immediately after the Power On Reset circuit drives the CDONE status pin LOW. The purpose of the initialization state is to clear all of the SRAM memory inside the FPGA.

The FPGA remains in the initialization state until the CRESETB pin is deasserted (HIGH) or until after the SSPI/SI^{2}C activation code is received.

4.3. Configuration Ports Default Behavior and Arbitration

During power up or when CRESETB pin toggles from LOW to HIGH, the Configuration Logic puts the device into Master Configuration Mode and boots either from internal NVCM or external SPI Flash.

The blank CrossLink device, without any software re-configuration, employs the default BOOT_UP_SEQUENCE of NVCM-EXT, a Dual Boot configuration mode. The configuration engine first attempts to boot from the NVCM. If the device fails to boot from NVCM, it attempts to boot from the external SPI Flash using the MSPI configuration mode.

Holding CRESETB LOW postpones the Master Configuration event, and allows the slave configuration ports (Slave SPI or Slave I^{2}C) to detect a 'Slave Active' condition.

1. During power up, CRESETB must be asserted (LOW) within 9.5 ms of Power-On-Reset supplies reaching the POR thresholds to prevent the CrossLink device from entering Master Configuration Mode
   a. CRESETB can be kept low prior to power up to prevent it from entering Master Configuration Mode.
   b. The activation keys can be send at any time while CRESETB is asserted (LOW).
2. While in User Mode, CRESETB can be toggled from HIGH to LOW at any time to re-enter Configuration Mode
   a. The activation key can be sent at any time while CRESETB is asserted (LOW).

Slave Active condition means that the slave port is addressed, the Slave Configuration Port Activation Key (as listed in Table 4.1) is sent, and the Activation Key matches the pre-defined key code. If any slave port declares active before CRESETB is released HIGH, the device is activated for slave configuration. The I^{2}C Slave Port declares itself active by asserting ACK after the final byte of the Activation Key written by the I^{2}C Master. However, in SPI mode, there is no mechanism for slave device to acknowledge the master device. If no Activation Key is received and CRESETB pin is released HIGH, the device enters Master Configuration Mode and boot from the location specified in the BOOT_UP_SEQUENCE preference.

Table 4.1. Slave Configuration Port Activation Key

Configuration Port	Header	Activation Key
Slave SPI Port	Dummy Bytes[1]	0xA4C6F48A
Slave I^{2}C Port	Slave I^{2}C Port Address Write[2]	0xA4C6F48A

Notes:
1. The number of dummy bytes should be at least 1.
2. The slave I^{2}C address could be either 7 bits or 10 bits address.

The I^{2}C and SPI pins are intentionally shared (MCK/SPI_SCK/SDA and CSN/SPI_SS/SCL) in such a manner as to prevent unintentional activation of either port. For example, a valid I^{2}C interface can never inadvertently activate the SPI port and vice versa.

4.4. Configuration

The FPGA is able to accept the configuration bitstream created by the Lattice Diamond development tools. The CrossLink device attempts to fetch configuration data from non-volatile memory, either the internal NVCM, an external SPI Flash, or both (Dual Boot). The configuration sequence is determined by BOOT_UP_SEQUENCE setting and the CrossLink device boots from the following sequence:

- The CrossLink device reads the BOOT_UP_SEQUENCE Feature Bits from the internal registers to determine the primary boot location. The hardware default value is NVCM-EXT, primary location is NVCM and secondary location is external SPI Flash (Master SPI).
- The CrossLink device attempts to boot from the primary boot location as specified in BOOT_UP_SEQUENCE (NVCM or external SPI Flash).
- If the CrossLink device fails to boot from the primary location and BOOT_UP_SEQUENCE specifies a secondary boot location, the device attempts to boot from this secondary boot location.
- If the CrossLink device fails to boot from both primary and secondary boot locations, CDONE signal remains low.
- If the CrossLink device does not find any valid configuration data, only the Slave SPI (SSPI) or Slave I^2C modes may be used to program the device. An external SPI Master or I^2C Master needs to write the Activation Key to the FPGA to enter one of these modes. See Configuration Ports Default Behavior and Arbitration section for additional details.

4.5. Wake-up

Wake-up is the transition from configuration mode to User Mode. The CrossLink device's fixed four-phase Wake-up sequence starts when the device has correctly received all of its configuration data. When all configuration data is received, the FPGA asserts an internal DONE status bit. The assertion of the internal DONE causes a Wake-up state machine to run that sequences four controls. The four control strobes are:

- External CDONE
- Global Write Disable (GWDIS)
- Global Output Enable (GOE)
- Global Set/Reset (GSR)

In the first phase of the Wake-up process, the CrossLink device releases the Global Output Enable and asserts the Global Write Disable.

When Global Output Enable is asserted, it permits the FPGA's I/O to exit a high-impedance state and take on their programmed output function. The FPGA inputs are always active. The input signals are prevented from performing any action on the FPGA flip-flops by the assertion of the Global Set/Reset (GSR).

The Global Write Disable is a control that overrides the write enable strobe for all RAM logic inside the FPGA. The inputs on the FPGA are always active, as mentioned in the Global Output Enable section. Keeping GWDIS asserted prevents accidental corruption of the instantiated RAM resources inside the FPGA.

The second phase of the Wake-up process releases the Global Set/Reset and the Global Write Disable controls.

The Global Set/Reset is an internal strobe that, when asserted, causes all I/O flip-flops, Look Up Table (LUT) flip-flops, distributed RAM output flip-flops, and Embedded Block RAM output flip-flops that have the *GSR enabled* attribute to be set/cleared per their hardware description language definition.

The last phase of the Wake-up process is to assert the external CDONE pin. The CDONE pin may also be held LOW externally to delay the User Mode entry in order to synchronize with other devices. This behavior is configurable, see the sysCONFIG Pins section for details on the CDONE pin.

When the final Wake-up phase is complete, the FPGA enters User Mode.

4.6. User Mode

The CrossLink device enters User Mode immediately after the Wake-up sequence has completed. User Mode is the point in time when the CrossLink device begins performing the logic operations in the design. The CrossLink device remains in this state until CRESETB is asserted or POR is triggered due to any of the power rails falling below the POR threshold level.

4.7. Clearing the Configuration Memory and Re-initialization

The current User Mode configuration of the CrossLink device remains in operation until it is actively cleared, or power is lost. Several methods are available to clear the internal configuration memory of the CrossLink device:

- Remove power so that power rails fall below the POR threshold.
- Execute an Erase command while in programming mode
- Toggle the CRESETB pin from HIGH to LOW. Note that only a HIGH to LOW transition creates a refresh event. Keeping CRESETB LOW does not create a refresh event.

Invoking one of these methods causes the CrossLink device to drive CDONE LOW. The CrossLink device enters the initialization state as described earlier.

4.8. Bitstream/Flash Sizes

The CrossLink device is an SRAM based FPGA. The SRAM configuration memory must be loaded from a non-volatile memory that can store all of the configuration data. The size of the configuration data is variable. It is based on the amount of logic available in the FPGA, and the number of pre-initialized Embedded Block RAM (EBR) components. A CrossLink design using the largest device, with every EBR pre-initialized with unique data values, and generated without compression turned on requires the largest amount of storage.

Table 4.2. Maximum Configuration Bits

Device	Bitstream Size Without Pre-Initialized EBR	Bitstream Size With Maximum Number of Pre-Initialized EBR	Units
LIF-MD6000	1.24	1.59	Mb

4.9. Configuration Modes of CrossLink

The CrossLink configuration SRAM memory must be loaded with valid configuration data before the FPGA operates. The CrossLink device provides four modes of loading the configuration data into the SRAM memory. The four modes available are Self-Download (NVCM) mode, Master SPI mode, Slave SPI mode, and Slave I^2C Configuration mode. When enabled in User Mode, these SPI and I^2C ports are dedicated for configuration only. They cannot be used as general purpose SPI port nor I^2C Port. The CrossLink device has two user I^2C IP cores, I2C0, and I2C1, for non-configuration purposes. For more details regarding the I^2C cores, see CrossLink Family Data Sheet (FPGA-DS-02007).
The Dual-boot operation is supported as a combination of the Self-Download mode and Master SPI mode.

4.10. sysCONFIG Pins

The CrossLink device provides a set of sysCONFIG I/O pins to program and configure the FPGA. The sysCONFIG pins are grouped together to create ports (SSPI, I^2C, MSPI) that are used to interact with the FPGA for programming, configuration, and access of resources inside the FPGA. The sysCONFIG pins in a configuration port group may be active, and used for programming the FPGA, or they can be reconfigured to act as general purpose I/O.

Recovering the configuration port pins for use as general purpose I/O requires adhering to the following guidelines:

- You must DISABLE the unused port. You can accomplish this by using the Diamond Spreadsheet View's Global Preferences tab. Each configuration port is listed in the sysCONFIG options tree.

Table 4.3 lists the default state of the shared sysCONFIG pins. A device with a blank/HW default Feature Row has the Master SPI port enabled. Upon entry to User Mode, the state of the sysCONFIG pins is determined by the sysCONFIG port settings. The SW default sysCONFIG port setting has the SSPI port enabled.

The sysCONFIG pins are powered by the VCCIO0 voltage. It is important that you take this into consideration when provisioning other logic attached to Bank 0.

The function of each sysCONFIG pin is described in Table 4.3.

Table 4.3. Default State of sysCONFIG Pins

Pin Name	Associated sysCONFIG Port	Pin Function in Configuration Mode	Pin Direction (Configuration Mode)	Software Default Function in User Mode
CRESETB	SDM	CRESETB	Input with weak pull-up	CRESETB
CDONE	SDM	I/O	I/O with weak pull-up	User-defined I/O
SPI_SCK/MCK/SDA	SSPI/MSPI/I²C	MSPI	Input with weak pull-up	SSPI
SPI_SS/CSN/SCL	SSPI/MSPI/I²C	MSPI	Input with weak pull-up	SSPI
MOSI	SSPI/MSPI	MSPI	Input	SSPI
MISO	SSPI/MSPI	MSPI	Output	SSPI

Note: All pins are in Configuration Mode until the device is configured and enters User Mode.

Table 4.4. Default State in Diamond for each Port

sysCONFIG Port	Diamond Default[1]
CDONE_PORT	CDONE_USER_IO
SLAVE_SPI_PORT	Enable
I2C_PORT	Disable
MASTER_SPI_PORT	Disable[2]

Notes:
1. This default setting can be modified in the Diamond Spreadsheet View, Global Preferences tab.
2. The MASTER_SPI_PORT setting does not influence the behavior during configuration. For details, see the Configuration section.

4.10.1. Self-Download Port Pins

CRESETB

The CRESETB is an active LOW input with a weak internal pull-up resistor used for configuration of the FPGA. When CRESETB is toggled from HIGH to LOW, the FPGA exits User Mode and starts a device configuration sequence at the Initialization phase, as described in Figure 4.1. Holding the CRESETB pin LOW during power up keeps the CrossLink device in the Initialization phase. This LOW period also allows an external SPI Master or I²C Master to write the Activation Key to the FPGA to enter into slave configuration mode. The CRESETB has a minimum pulse width assertion period in order for it to be recognized by the FPGA. You can find this minimum time in CrossLink Family Data Sheet (FPGA-DS-02007) in the AC timing section.

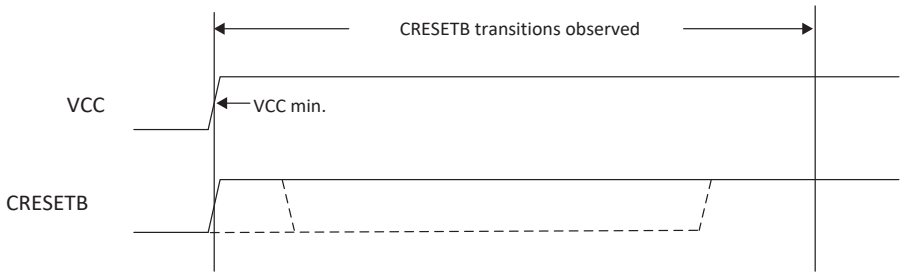

Figure 4.2. Period CRESETB is Always Observed

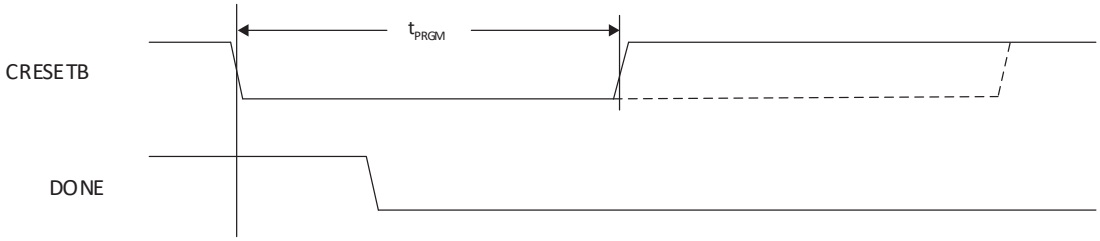

Figure 4.3. Configuration from CRESETB Timing

If an error is detected when reading the bitstream, the internal DONE bit is not set, the CDONE pin stays LOW, and the device does not wake up. The device configuration fails when the following occurs:

- The bitstream CRC error is detected.
- The invalid command error is detected.
- A timeout error is encountered when loading from the on-chip NVCM.
- The program DONE command is not received when the end of on-chip SRAM configuration or on-chip NVCM is reached.
- Device ID code mismatch.

CDONE

The CDONE pin is a bi-directional open drain with a weak pull-up that signals the FPGA is in User mode. CDONE is first able to indicate entry into User Mode only after an internal DONE bit is asserted. The internal DONE bit defines the beginning of the FPGA Wake-up state.

The CDONE output pin is controlled by the CDONE_PORT and DONE_EX configuration parameters that can be modified in the Diamond Spreadsheet View. By default, the CDONE pin is a general purpose I/O when the CrossLink device is in the Feature Row HW Default Mode state. The default mode causes the CrossLink device to automatically pass through the Wake-up sequence after the internal DONE bit is asserted. The FPGA does not stall waking up waiting for the CDONE pin to be asserted high.

The FPGA can be held from entering User Mode indefinitely by having an external agent keep the CDONE pin asserted LOW. In order to use CDONE to delay the FPGA from entering User Mode, the CDONE_PORT must be set to CDONE_ONLY and the DONE_EX set to ON. These setting can be modified from Diamond Spread Sheet View. With this setting, the device waits for CDONE to be driven HIGH by an external signal to wake up. A common reason for keeping CDONE driven LOW is to allow multiple FPGAs to be completely configured. As each FPGA reaches the DONE state, it is ready to begin operation. The last FPGA to configure can cause all FPGAs to start in unison.

The CDONE pin drives LOW when the FPGA enters Initialization mode. As described earlier, this condition happens when power is applied, CRESETB is asserted through HIGH to LOW transition.

4.10.2. Master and Slave SPI Configuration Port Pins

Table 4.5. Master SPI Configuration Port Pins

Pin Name	Function	Direction	Description
MCK	MCK	Output with weak pull-up	Master clock used to time data transmission/reception from the CrossLink Configuration Logic to a slave SPI Flash.
CRESETB	CRESETB	Input with weak pull-up	CRESETB is used to initiate programming / configuration
MOSI	MO	Output	This is the Master output which carries configuration commands to the external SPI Flash.
MISO	MI	Input	This is the input to the Master which carries output data from the slave SPI Flash to the CrossLink Configuration Logic.
CSN	CSN	Output	CrossLink Master SPI chip select output pin for external SPI Flash.

Table 4.6. Slave SPI Configuration Port Pins

Pin Name	Function	Direction	Description
SPI_SCK	SPI_SCK	Input with weak pull-up	Clock used to time data transmission/reception from an external SPI master device to the CrossLink Configuration Logic.
CRESETB	CRESETB	Input with weak pull-up	CRESETB is used to initiate programming / configuration
MOSI	SI	Input	This is the input to the Slave which receives data from the external SPI Master to the CrossLink Configuration Logic.
MISO	SO	Output	This is the output from the Slave which carries output data from the CrossLink Configuration Logic to the external SPI Master.
SPI_SS	SPI_SS	Input with weak pull-up	CrossLink Configuration Logic slave SPI chip select input.

MCK

The MCK pin is the Master SPI port's clock pin. The MCK pin is available when the CrossLink device is in Feature Row HW Default mode, in Master SPI configuration mode, or in User Mode when the Master SPI port is enabled through Diamond Spreadsheet View. MCK becomes an output and provides a reference clock for the SPI Flash attached to the CrossLink device's Master SPI Configuration port. MCK actively drives out a clock until all of the configuration data has been received from the external SPI Flash.

The CrossLink device generates MCK from an internal oscillator. The initial frequency of MCK is nominally 6 MHz. The MCK frequency can be altered using the MCCLK_FREQ parameter in the Diamond Spreadsheet View, Global Preferences tab. Following MCCLK frequencies are supported:

- 1 MHz
- 2 MHz
- 3 MHz
- 6 MHz
- 12 MHz
- 24 MHz

The MCCLK_FREQ value is loaded into the FPGA during the initial stages of device configuration. After the CrossLink device accepts the new MCCLK_FREQ value, the MCK output begins driving the selected frequency. When selecting the MCCLK_FREQ, ensure that you do not exceed the frequency specification of your configuration memory, or of your PCB.

SPI_SCK

The SPI_SCK pin is the Slave SPI port's clock pin. The SPI_SCK pin is available when the CrossLink device is in Slave SPI configuration mode or in User Mode when the Slave SPI port is enabled through Diamond Spreadsheet View. When the Slave SPI port is enabled, it allows an external SPI Master controller to drive the clock to this pin and program the configuration bitstream to the CrossLink device. Refer to the CrossLink AC specifications in the CrossLink Family Data Sheet (FPGA-DS-02007) for the maximum SPI_SCK frequency and data setup/hold parameters.

Note: In the CrossLink device, the SPI_SCK pin is shared with the configuration I^2C SDA line. The startup sequence identifies if Slave SPI/I^2C mode is activated. When one configuration port is activated, the other configuration port is locked out until the next time the device goes through the initialization state (see Figure 4.1 for details).

SPI_SS

The SPI_SS pin is the Slave SPI port's chip select pin. The SPI_SS pin is available when the CrossLink device is in Slave SPI configuration mode or in User Mode when the Slave SPI port is enabled through Diamond Spreadsheet View. When the Slave SPI port is enabled, it allows an external SPI Master to assert the SPI_SS pin (LOW) in order to write the configuration bitstream to the CrossLink device.

Proper operation of the CrossLink device depends upon maintaining the SPI_SS pin in the correct state:

- Care should be taken to minimize the noise on SPI_SS signal during power up. Noise on SPI_SS pin may cause the SPI Master to fail during the writing of the configuration bitstream to CrossLInk. SPI_SS must be asserted when configuring using Slave SPI mode.
- When the Slave SPI port is enabled in User Mode, the Master SPI device should keep SPI_SS deasserted (HIGH) until the SPI Master is ready to access it to prevent any spurious access.
    - The Master SPI port and the Slave SPI port share the same pins. If both ports are enabled at the same time, Diamond flow fails.

Lattice recommends adding a 4.7 kΩ pull-up resistor to the SPI_SS pin to augment the weak internal pull-up.

Note: In the CrossLink device, the SPI_SS pin is shared with the configuration I^2C SCL line. The startup sequence identifies if Slave SPI/I^2C mode is activated. When one configuration port is activated, the other configuration port is locked out until the power cycle.

CSN

The CSN pin (active LOW) is the Master SPI port's chip select pin. The CSN pin is available when the CrossLink device is in Feature Row HW Default mode, in Master SPI configuration mode, or in User Mode when the Master SPI port is enabled through Diamond Spreadsheet View. The CrossLink device asserts the CSN pin when attempting to boot through Master SPI mode. The CrossLink device asserts CSN until all configuration data bytes have been loaded, at which time the MCK enters a high impedance state.

The CSN/SPI_SS pin remains in SPI_SS mode after the FPGA enters User Mode when left in the software default state. You must ENABLE the Master SPI port and DISABLE the Slave SPI port through Diamond Spreadsheet View to preserve the CSN functionality in User Mode.

Notes:

- When configuring the CrossLink device from an external SPI Flash, ensure the SPI Flash V_{CC} and the CrossLink device's V_{CCIO0} are at the same level by connecting them to the same power source. (Some SPI PROM manufacturers may require the chip select input of the PROM ramp in unison to the PROM's V_{CC} rail.)
- Ensure that the SPI Flash V_{CC} is at the recommended operating level.
- The CSN/SPI_SS pin, by default, has a weak internal pull-up resistor.

MOSI

MOSI is a dual function bi-directional pin. The direction depends upon whether the Master or Slave mode is active. The MOSI is an input data pin when using the Slave SPI mode and is an output data pin when using the Master SPI mode. In Master SPI mode, the CrossLink device drives MOSI until all configuration data bytes have been loaded, at which time the MOSI enters a high impedance state.

At least one of the sysCONFIG preferences, MASTER_SPI_PORT or SLAVE_SPI_PORT, must be set to ENABLE in order to preserve this pin as MOSI and allow access to the SPI interface.

MISO

MISO is a dual function bi-directional pin. The direction depends upon whether a Master or Slave mode is active. The MISO is an input data pin when using the Master SPI mode and is an output data pin when using the Slave SPI mode.

At least one of the sysCONFIG preferences, MASTER_SPI_PORT or SLAVE_SPI_PORT, must be set to ENABLE in order to preserve this pin as MISO and allow access to the SPI interface.

CRESETB

CRESETB is a configuration reset pin. When CRESETB is asserted through a HIGH to LOW transition, the FPGA exits User Mode and starts a device configuration sequence at the Initialization phase, as described in this Tech Note. Holding the CRESETB pin LOW prevents the CrossLink device from leaving the Initialization phase. An external SPI Master or I^2C Master can also write the Activation Key to the FPGA during this LOW time to enter slave configuration mode.

4.10.3. I^2C Configuration Port Pins

SCL

The CrossLink device provides an I^2C configuration port. The SCL is the bi-directional I^2C Serial Clock pin, and is used to initiate and time transactions on the I^2C bus. SCL requires an external pull-up resistor in order to operate.

The SCL pin is configured as CSN when the CrossLink device is in the Feature Row HW Default Mode state. This pin takes on the SCL functionality if a Master I^2C device writes the I^2C activation key to the CrossLink device while in Configuration Mode or ENABLE the I2C_PORT through Diamond Spreadsheet View to enable the I^2C configuration port in User Mode (see the I2C Configuration Mode section for details). The I2C0 and I2C1 User Mode I^2C blocks operate independently of the configuration I^2C block. The configuration SCL pin is not shared with the I2C0 USER_SCL pin nor the I2C1 SCL pin.

SDA

The SDA pin is the I^2C serial data input/output pin. It is bidirectional, open-drain, and requires an external pull-up resistor in order to operate. The pin changes direction dynamically during data transactions on the I^2C bus. The current state depends on the current bus master and the operation being performed by that master.

The SDA pin is configured as MCK when the CrossLink device is in the Feature Row HW Default Mode state. This pin takes on the SDA functionality if a Master I^2C device writes the I^2C activation key to the CrossLink device while in Configuration Mode or ENABLE the I2C_PORT through Diamond Spreadsheet View to enable the I^2C configuration port in User Mode (see the I2C Configuration Mode section for details). The I2C0 and I2C1 User Mode I^2C blocks operate independently of the configuration I^2C block. The configuration SDA pin is not shared with the I2C0 USER_SDA pin nor the I2C1 SDA pin.

5. Configuration Modes

The CrossLink device provides multiple options for loading the configuration SRAM from a non-volatile memory. The previous section describes the physical interface necessary to interact with the CrossLink Configuration Logic. This section describes the functionality of each of the different configuration modes. Descriptions of important settings required in the Diamond Spreadsheet View are also discussed. See the Configuration section for details on the default configuration behavior of the device.

5.1. SDM Mode

SDM (Self-Download Mode) is the primary configuration method for the CrossLink device. The advantages of SDM include:

- Speed – The CrossLink device is ready to run in a few milliseconds depending on the density of the device.
- Security – The configuration data is never seen outside the device during the load to SRAM. You can also prevent the internal memory from being read.
- Reduced cost – There is no need to purchase a Flash specifically reserved for programming the CrossLink device.
- Reduced board space – Elimination of an external Flash allows your board to be smaller.

The CrossLink device retrieves the configuration data from the internal NVCM when it is using Self-Download Mode. SDM is triggered when power is applied or by asserting the CRESETB pin from HIGH to LOW.

5.2. Master SPI Configuration Mode

Master SPI (MSPI) configuration mode is the only other self-controlled configuration mode available to the CrossLink device. Lattice recommends having a secondary configuration port available. The secondary port allows you to recover the CrossLink device in the event of a programming error.

For the CrossLink device to operate correctly using the MSPI configuration mode, ensure that:

- The POR of the SPI Flash device is lower than the POR of the CrossLink device or the SPI Flash is powered first.
- SPI Flash Fmax is greater than the selected MCCLK_FREQ setting (default = 6 MHz).

Table 5.1. Master SPI Configuration Port Pins

Pin Name	Function
MCK	Clock output from the CrossLink Configuration Logic and Master SPI controller. Connect MLK to the SCLK input of the Slave SPI device.
MOSI	Serial Data output from the CrossLink device to the slave SPI SI input.
MISO	Serial Data input to the CrossLink Configuration Logic from the slave SPI SO output.
CSN	Chip select output from the CrossLink Configuration Logic to the slave SPI Flash holding configuration data for the CrossLink device.

Table 4.2 provides information about the amount of memory needed for CrossLink configuration data. Select an SPI Flash that accepts 03 hex Read Opcodes. The CrossLink device is only able to use the 03 hex Read Opcode.

When the CrossLink device is in Master SPI configuration mode, the MCK/SPI_SCK I/O takes on the Master Clock (MCK) function, and begins driving a nominal 6 MHz clock to the SPI Flash's SCLK input. CSN is asserted LOW, commands are transmitted to the Flash over the MOSI output, and data is read from the Flash on the MISO input pin. When all of the configuration data is retrieved from the Flash, the CSN pin is deasserted and the MSPI output pins are tristated.

The MCCLK_FREQ default value is 6 MHz. The MCCLK_FREQ parameter, accessed using Diamond Spreadsheet View, can be used to increase or decrease the configuration frequency.

The configuration data in the Flash has some padding bits, and then the data altering the MCK base frequency is read. The CrossLink device reads the remaining configuration data bytes using the new MCK frequency.

After the CrossLink device enters User Mode, the Master SPI configuration port pins tristate. This permits background programming of or access to the SPI Flash.

To set the CrossLink device for operation using the MSPI configuration mode:

1. Store the entire configuration data in an external SPI Flash.
2. The data must start at offset 0x000000 within the Flash.
3. Set the preferences as listed in Table 5.2.
4. Enable Bitstream File creation in the Diamond Process Pane.
5. Run the Export Files process to build your design.

Table 5.2. Master SPI Configuration Software Settings

Preference	Setting
MASTER_SPI_PORT (in User Mode)	ENABLE
BOOT_UP_SEQUENCE	EXT, NVCM-EXT, EXT-NVCM, or EXT-EXT

The Export Files process generates a PROM file, a JEDEC file, and/or a BIT file. The BIT file must be programmed into the external SPI Flash. There are several ways to get the data into the SPI Flash:

- Diamond Programmer can transmit the SPI Flash data using a download cable
- An on-board SOC can program the SPI Flash
- Automatic test equipment can program the SPI Flash
- Pre-programmed SPI Flash memories can be pre-assembled onto your printed-circuit board

When the SPI Flash contains the configuration data, you can test the configuration. Toggle the CRESETB pin through HIGH to LOW to HIGH transition or cycle power to the board, and the CrossLink device is configured from the external SPI Flash.

After the CrossLink device enters User Mode, the MCK pin may retain the MSPI functionality or become a general purpose I/O depending on the setting in Diamond Spreadsheet View under Global Preferences tab.

5.3. Dual Boot Configuration Mode

Dual Boot Configuration Mode is a combination of Self Download Mode and Master SPI Configuration Mode. When set up in Dual Boot Mode, the CrossLink device tries to configure first from a primary image stored in external SPI Flash or NVCM. If the primary image configuration fails, the CrossLink device attempts to configure itself using a failsafe golden image stored in either external SPI Flash or NVCM. The load order can be changed by setting the BOOT_UP_SEQUENCE preference.

The primary image can fail in one of several ways:

- A bitstream CRC error is detected during configuration
- A time-out error is encountered when loading the configuration SRAM
- A Device ID mismatch occurs during configuration
- An illegal command is asserted which can cause failure

A CRC error is caused by incorrect data being written into the internal NVCM or external SPI Flash. The configuration data is read out in rows. As each row enters the Configuration Engine the data is checked for CRC consistency. Before the data enters the Configuration SRAM the CRC must be correct. Any incorrect CRC causes the device to erase the Configuration SRAM and retrieve configuration data from the failsafe image.

There is a corner case wherein it is possible for the data to be correct from a CRC calculation perspective, but not functionally correct. In this instance, the internal DONE bit never becomes active. The CrossLink device counts the number of master clock pulses it provided after the Power On Reset signal is released. When the count expires without DONE becoming active and DONE_EX = OFF, the FPGA attempts to get its configuration data from the failsafe image.

The external SPI Flash must have a lower Power-On-Reset voltage supply level than the CrossLink POR to ensure proper configuration.

Dual boot configuration mode typically requires two configuration data files. One of the two configuration data files is a failsafe image that is rarely, if ever, updated. The second configuration data file is a working image (also called primary image) that is routinely updated. One Diamond project (or implementation) can be used to create both the working and the failsafe configuration data files.

Refer to the Diamond Online Help for more information about using Diamond implementations.

Use the following preferences to build a dual-boot design:

Table 5.3. Dual-Boot Configuration Settings

Preference	Dual-Boot Setting
MASTER_SPI_PORT (in User Mode)	ENABLED
BOOT_UP_SEQUENCE	NVCM-EXT, EXT-NVCM, EXT-EXT

The failsafe configuration image generated by Diamond can be stored in either the NVCM or the external SPI Flash. In dual boot scenarios where only one image is stored in the external SPI Flash (BOOT_UP_SEQUENCE = NVCM-EXT or EXT-NVCM), the external image is stored in the SPI Flash starting at address 0x000000. This differs from a single image Master SPI Configuration Mode (BOOT_UP_SEQUENCE = EXT-EXT), which requires the primary configuration data to be stored at offset 0x000000 and the secondary configuration data can be configured through Diamond Deployment Tool. The external SPI Flash memory file for dual boot can be generated using the Diamond Deployment Tool. Use the **External Memory: Dual Boot** option in Deployment Tool to generate the dual boot image.

5.4. Slave SPI Mode

The CrossLink device provides a Slave SPI (SSPI) configuration port that allows you to access features provided by the Configuration Logic. To enter SSPI mode, CRESETB pin should be held LOW while an external SPI Master writes the Activation Key (Table 4.1) to the FPGA. See Configuration Ports Default Behavior and Arbitration section for additional details.

Table 5.4. Slave SPI Port Pins

Pin Name	Description
SPI_SCK	Configuration clock input that is driven by a SPI master controller.
MOSI	Serial Data Input to the CrossLink Configuration Logic for command and data.
MISO	Serial Data Output from the CrossLink Configuration Logic.
SPI_SS	Chip select to enable the CrossLink Configuration Logic.

In the Slave SPI mode, the MCK/SPI_SCK pin becomes SPI_SCK (that is Configuration clock). Input data is read into the CrossLink device on the MOSI pin at the rising edge of SPI_SCK. Output data is valid on the MISO pin at the falling edge of SPI_SCK. The SPI_SS acts as the chip select signal. When SPI_SS is high, the SSPI interface is deselected. The state of the SPI pins are shown in Table 5.5.

Table 5.5. Slave SPI Port Pin States

Pin Name	SPI_SS = HIGH	SPI_SS = LOW
SPI_SCK	Tri-state	Input
MOSI	Tri-state	Input
MISO	Output*	Output

***Note:** Due to this limitation, multiple Slave SPI devices cannot be supported on the same SPI bus which the CrossLink device resides.

Commands can be written into and data read from the CrossLink device when SPI_SS is asserted. The CrossLink SSPI port only accepts Mode 0 bus transactions to the Configuration Logic, Where the Mode 0 bus transaction is the Master SPI setting of configuration master configured at CPHA = 0 and CPOL = 0.

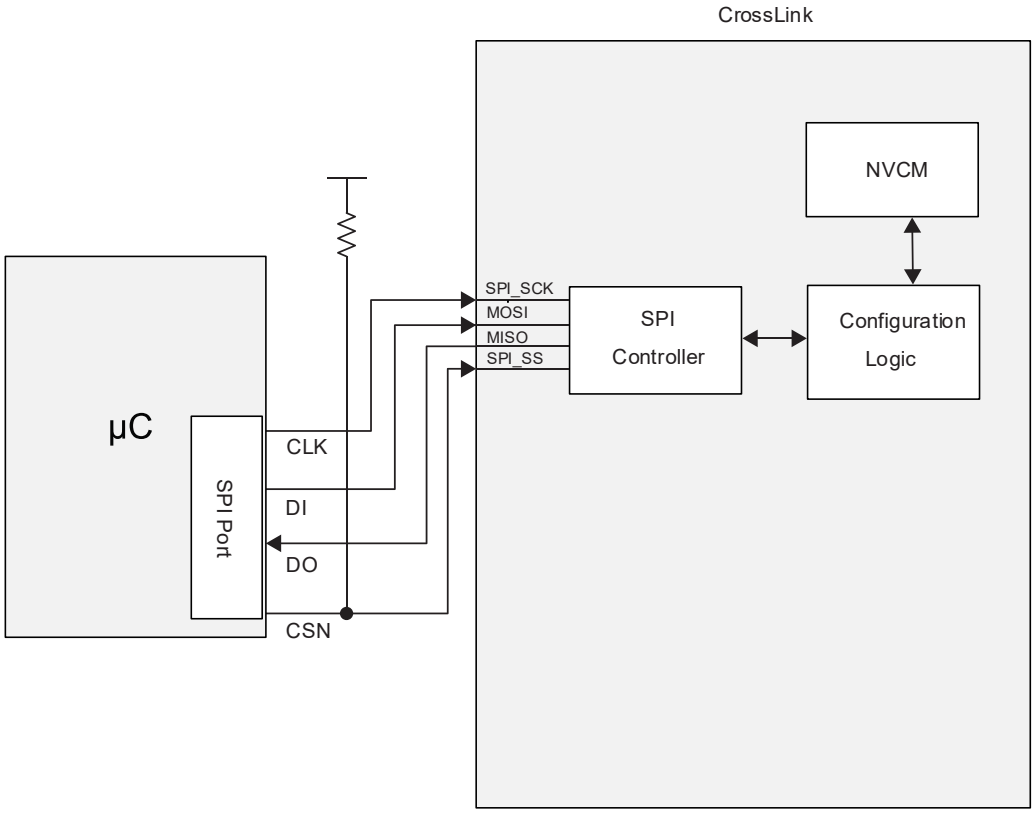

Figure 5.1. Slave SPI Configuration Mode

The SSPI port is used to program and verify the NVCM or to configure the SRAM. Programming CrossLink using the SSPI port is complex. Lattice provides *C* source code called SSPIEmbedded to circumvent the complexity of programming CrossLink. Use SSPIEmbedded to reprogram the CrossLink NVCM. To modify the SSPIEmbedded code as per the user-specific environment programming master, refer to the Programming Tools User Guide document.

5.5. I²C Configuration Mode

The CrossLink device has an I²C Configuration port for use in accessing the Configuration Logic. An I²C Master can access the Configuration Logic using address 1000000 (7-bit mode) or 1111000000 (10-bit mode) unless the I²C base address has been modified. You can program the NVCM (one-time programmable) or configure the SRAM and access status/control registers within the Configuration Logic block. Note that only one of the SPI or I²C interfaces can be operated at a time. When the I²C interface is activated, all subsequent communication to the SPI port is ignored.

Table 5.6. I²C Port Pins

Pin Name	Description
SCL	I²C bus clock
SDA	I²C bus data line

The default state set for the I2C_PORT in the Diamond design software is to place the I2C_PORT in the DISABLE state. You must make sure the I2C_PORT is set to the ENABLE state to leave the I²C interface active in User Mode.

To enter Slave I²C mode, CRESETB pin should be held LOW while an external I²C Master writes the Activation Key (see Table 4.1) to the FPGA. See Configuration Ports Default Behavior and Arbitration section for additional details.

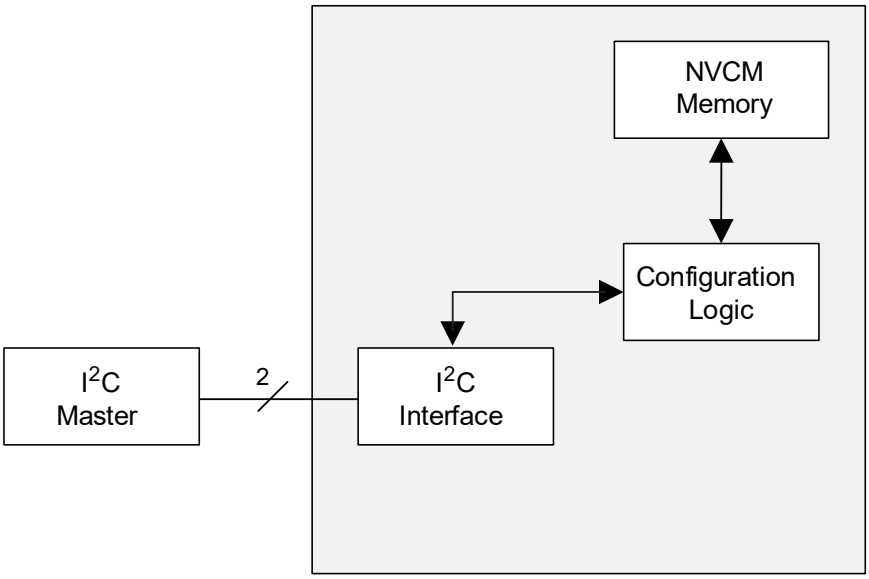

Figure 5.2. I²C Configuration Logic

Table 5.7 lists the address decoding used to access the I²C resources in the CrossLink device.

Table 5.7. Slave Addresses for I²C Ports

Slave Address	I²C Function
yyyxxxxx00b	Primary I²C Controller Configuration Logic address. Always responds to 7-bit or 10-bit addresses.
yyyxxxxx11b	Primary I²C Configuration Logic Reset. Always responds to 7-bit or 10-bit addresses.

Note: Although there are four possible combinations of the reserved address bits 1000 0XX, only the two combinations listed above are used. The remaining two addresses are reserved for future I²C bus enhancements.

The CrossLink I²C controller supports two separate slave addresses as listed in Table 5.7. These are determined by the two least significant bits in the slave address – 00 corresponds to the Configuration Logic, while 11 corresponds to a reset port. In some instances, an I²C memory transaction to the Configuration Logic may be interrupted or abandoned.

It is possible for a command to be accepted by the Configuration Logic that causes the Configuration Logic to respond with data. In the event that the I²C memory transaction is interrupted or abandoned, the Configuration Logic continues to return the queued data.

New incoming I²C commands may be considered padding bytes or may be misinterpreted. Clear this condition by writing any value to the address with least significant bits 11. The Configuration Logic command interpreter is reset, any queued data is flushed, and subsequent I²C memory transactions to the Configuration Logic operates correctly.

6. Software Selectable Options

The operation of the CrossLink Configuration Logic is managed by options selected in the Diamond design software. Other FPGAs provide dedicated I/O pins to select the configuration mode. The CrossLink device uses the non-volatile Feature Row to select how it configures. The Feature Row's default state needs to be modified in almost every design. Use the Diamond Spreadsheet View to make the changes to the operation of the CrossLink Feature Row which alters the operation of the Configuration Logic.

The Configuration Logic preferences are accessed using Spreadsheet View. Click the Global Preferences tab and look for the sysCONFIG tree. Figure 6.1 shows the sysCONFIG section. The sysCONFIG preferences are divided into three categories:

- Configuration mode and port related
- Bitstream generation related
- Security related

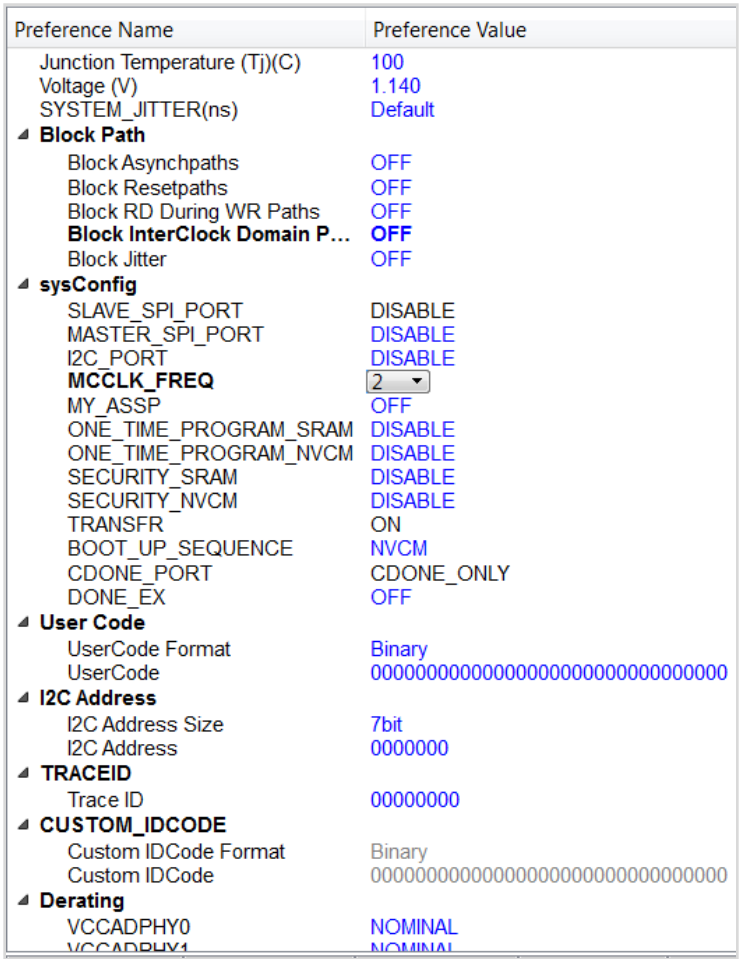

Figure 6.1. sysCONFIG Preferences in Global Preferences Tab, Diamond Spreadsheet View

6.1. Configuration Mode and Port Options

The configuration and port options allow you to set which configuration ports continue to operate after the CrossLink device enters User Mode. This only applies to User Mode and does not affect the programming of CrossLink during configuration. You can also control the availability of status pins, as well as the speed at which configuration data is read from an external Flash.

The configuration and port options can be used in any combination.

Table 6.1. Configuration Mode/Port Options

Option Name	Default Setting	All Settings
SLAVE_SPI_PORT	ENABLE	DISABLE, ENABLE
MASTER_SPI_PORT	DISABLE	DISABLE, ENABLE
I2C_PORT	DISABLE	DISABLE, ENABLE
MCCLK_FREQ	6	See description below
TRANSFR	OFF	OFF,ON
CDONE_PORT	CDONE_USER_IO	CDONE_USER_IO, CDONE_ONLY
DONE_EX	OFF	OFF, ON

SLAVE_SPI_PORT

The SLAVE_SPI_PORT allows you to preserve the Slave SPI configuration port after the CrossLink device enters User Mode. The SLAVE_SPI_PORT preference can be set in two states:

- ENABLE — This setting preserves the Slave SPI port I/O when the CrossLink device is in User Mode. When the pins are preserved, an external SPI master controller can interact with the Configuration Logic. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- DISABLE — This setting disconnects the SPI port pins from the Configuration Logic. By itself it does not make the port pins general purpose I/O. Which port pins becomes general purpose I/O depends on SLAVE_SPI_PORT, MASTER_SPI_PORT, and I2C_PORT settings.

MASTER_SPI_PORT

The MASTER_SPI_PORT allows you to preserve the Master SPI configuration port after the CrossLink device enters User Mode. The MASTER_SPI_PORT preference can be set in two states:

- ENABLE — This setting preserves the Master SPI port I/O when the CrossLink device is in User Mode. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- DISABLE — This setting disconnects the SPI port pins from the Configuration Logic. By itself it does not make the port pins general purpose I/O. Which port pins becomes general purpose I/O depends on SLAVE_SPI_PORT, MASTER_SPI_PORT, and I2C_PORT settings.

I2C_PORT

The I2C_PORT allows you to preserve the I^2C configuration port after the CrossLink device enters User Mode. The I2C_PORT preference can be set in two states:

- ENABLE — This setting preserves the I^2C port I/O when the CrossLink device is in User Mode. Choosing ENABLE also prevents you from over-assigning I/O to the port pins.
- DISABLE — This setting disconnects the I^2C port pins from the Configuration Logic. Which port pins becomes general purpose I/O depends on SLAVE_SPI_PORT, MASTER_SPI_PORT, and I2C_PORT settings.

MCCLK_FREQ

The MCCLK_FREQ preference allows you to alter the MCCLK frequency used to retrieve data from an external SPI Flash when using EXTERNAL or Dual Boot configuration modes. The MCCLK_FREQ value is stored in the incoming configuration data. It is not stored in the Feature Row. The CrossLink device reads a series of padding bits, a *start of data* word (0xBDB3) and a control register value. The control register contains the new MCCLK_FREQ value. CrossLink switches to the new clock frequency shortly after receiving the MCCLK_FREQ value.

The MCCLK_FREQ has a range of possible frequencies available from 1 MHz up to 24 MHz. Do not exceed the maximum clock rate of your SPI Flash, or of your printed circuit board.

Lattice recommends having a back-up configuration port available in case you specify a clock frequency that is out of specification.

TRANSFR

Leave at default setting of OFF.

CDONE_PORT

CDONE and a GPIO are bonded to a shared package ball. The ball configuration can be selected through Spreadsheet View. If CDONE_PORT = CDONE_ONLY, the GPIO becomes unavailable to you. If CDONE_PORT = CDONE_USER_IO, the GPIO becomes available to you.

DONE_EX

If ON, the device waits for CDONE to be driven high by an external signal to wake up. A common reason for keeping CDONE driven LOW is to allow multiple FPGAs to be completely configured. As each FPGA reaches the DONE state, it is ready to begin operation. The last FPGA to configure can cause all FPGAs to start in unison.

The default setting is OFF. The CDONE pin is asserted by the device when the internal DONE bit is set. When set to OFF, the device wakes up regardless of the external signal state on CDONE.

6.2. Bitstream Generation Options

The Bitstream Generation options allow you to decide how the Diamond development tools create the configuration data for the CrossLink device. The CONFIGURATION, USERCODE, and CUSTOM_IDCODE settings are saved in the Feature Row. The other options allow you to control the BIT files that are generated by Diamond.

CONFIGURATION

The BOOT_UP_SEQUENCE preference allows you to control the boot up sequence. The BOOT_UP_SEQUENCE preference has five possible settings:

- NVCM — The NVCM setting is the SW default mode for building configuration data. The configuration bitstream is stored in the Configuration NVCM.
- NVCM-EXT — This setting boots up the system using the NVCM first. If an error occurs, the system boots up with the golden image in the External SPI Flash. This is the HW default mode for a blank device.
- EXT-NVCM — This setting boots up the system using the External SPI Flash first. If an error occurs, the system boots up with the golden image in the NVCM.
- EXT-EXT — This setting enables you to use the external SPI Flash itself to store two configuration images. This setting boots up the system with the image kept at the first sector of the SPI Flash. If the system fails to boot with the first image, the system boots up with the second golden image at the second sector.
- EXT — This preference generates configuration data that is stored in an external Flash memory.

UserCode

The CrossLink Configuration NVCM sector contains a 32-bit register for storing a user-defined value. The default value stored in the register is 0x00000000. The UserCode preference allows you to assign any value to the register. Suggested uses include the configuration data version number, a manufacturing ID code, and date of assembly among others.

The format of the UserCode field is controlled using the UserCode Format preference. Data entry can be performed in either Binary, Hex, or ASCII formats.

UserCode Format

The UserCode Format preference sets the format for the data field used to assign a value in the USERCODE preference. The UserCode Format has three options:

- Binary — UserCode is set using 32 *1* or *0* characters.
- Hex — UserCode is set using eight hexadecimal digits (that is 0-9A-F)

- ASCII — UserCode is set using up to four ASCII characters

6.3. Security Options

Security Options allows you to select from a range of options for tracking or securing the CrossLink device. Table 6.2 provides a summary of these options.

Table 6.2. Configuration Mode/Port Options

Option Name	Default Setting	All Settings
Trace ID	<all zeros>	8-bit arbitrary
MY_ASSP	OFF	OFF, ON
SECURITY_NVCM	DISABLE	DISABLE, ENABLE
SECURITY_SRAM	DISABLE	DISABLE, ENABLE
ONE_TIME_PROGRAM_SRAM	DISABLE	DISABLE, ENABLE
ONE_TIME_PROGRAM_NVCM	DISABLE	DISABLE, ENABLE

Trace ID

Trace ID stamps each CrossLink with a unique 64-bit ID. No two CrossLink devices have the same Trace ID value even when they are loaded with the same configuration data. This differs from a UserCode which is present in the configuration data. Every device that receives the configuration data using a UserCode receives the same UserCode value.

The Trace ID is 64 bits long with the least significant 56 bits being immutable data. The 56 bits are a combination of the wafer lot, the wafer number and the X/Y coordinates locating the die on the wafer. The most significant eight bits are provided by you and are stored in the Feature Row. The Trace ID is changed using the Diamond Spreadsheet View. You enter a unique 8-bit binary value in the Trace ID field and generate configuration data.

MY_ASSP

Every Lattice device has its own identification code identifying the device family, device density, and other parameters (for example, voltage, device stepping, and so on). The code is accessible from any CrossLink configuration port. The value stored in the Custom IDCode register allows you to uniquely identify a Lattice device.

The MY_ASSP preference permits you to change the value returned when the Custom IDCode is read from the FPGA. Turning the MY_ASSP ON enables the Custom IDCode preference.

Custom IDCode

The Custom IDCode is the value you assign to override the default IDCode in the CrossLink device. You are only allowed to enter a 32-bit hexadecimal or binary value when the MY_ASSP preference is ON.

Overriding the IDCode prevents the Lattice programming software from being able to identify the CrossLink device, and as a result, prevents Programmer from being able to directly program the CrossLink device. It is necessary to migrate to generating Serial Vector Format (SVF) files in order to program MY_ASSP enabled CrossLink devices.

Custom IDCode Format

The Custom IDCode Format preference selects the format for the data field used to assign a value in the Custom IDCode preference. The Custom IDCode Format has three options:

- Binary — Custom IDCode is set using 32 *1* or *0* characters.
- Hex — Custom IDCode is set using eight hexadecimal digits (that is 0-9A-F).
- ASCII – Custom IDCode is set using up to four ASCII characters.

SECURITY_SRAM and SECURITY_NVCM

When these preferences are set to ON, the read-back of the SRAM memory or the NVCM are blocked. The CrossLink device cannot be read back.

ONE_TIME_PROGRAM_SRAM and ONE_TIME_PROGRAM_NVCM

ONE_TIME_PROGRAM_SRAM can be used to prevent the SRAM from being erased or reprogrammed. Leave ONE_TIME_PROGRAM_NVCM at the default (DISABLE) value. ONE_TIME_PROGRAM_NVCM has no affect since CrossLink NVCM is one time programmable.

7. Feature Row

The CrossLink device includes a Feature Row that is used to control FPGA resources. The Feature Row permits more flexibility in selecting the functions available for configuration. The Feature Row is stored in the NVCM, therefore, it is one-time programmable.

The CrossLink Feature Row can be used to determine how the device's SRAM configuration memory is loaded. When Feature Row is blank, the default configuration setting is NVCM-EXT. Feature Row can be read and modified using Programming File Utility under *Tools > Feature Row Editor*. The Feature Row can also be read through Diamond Programmer by selecting *Advanced Security Keys Programming* under *Access Mode* and *Security Read Feature Rows* under *Operation* (Figure 7.2). The Feature Row can be programmed thorough Diamond Programmer by selecting *SSPI NVCM Programming* under *Access Mode* and *SSPI NVCM Program Feature Rows* under *Operation* (Figure 7.3).

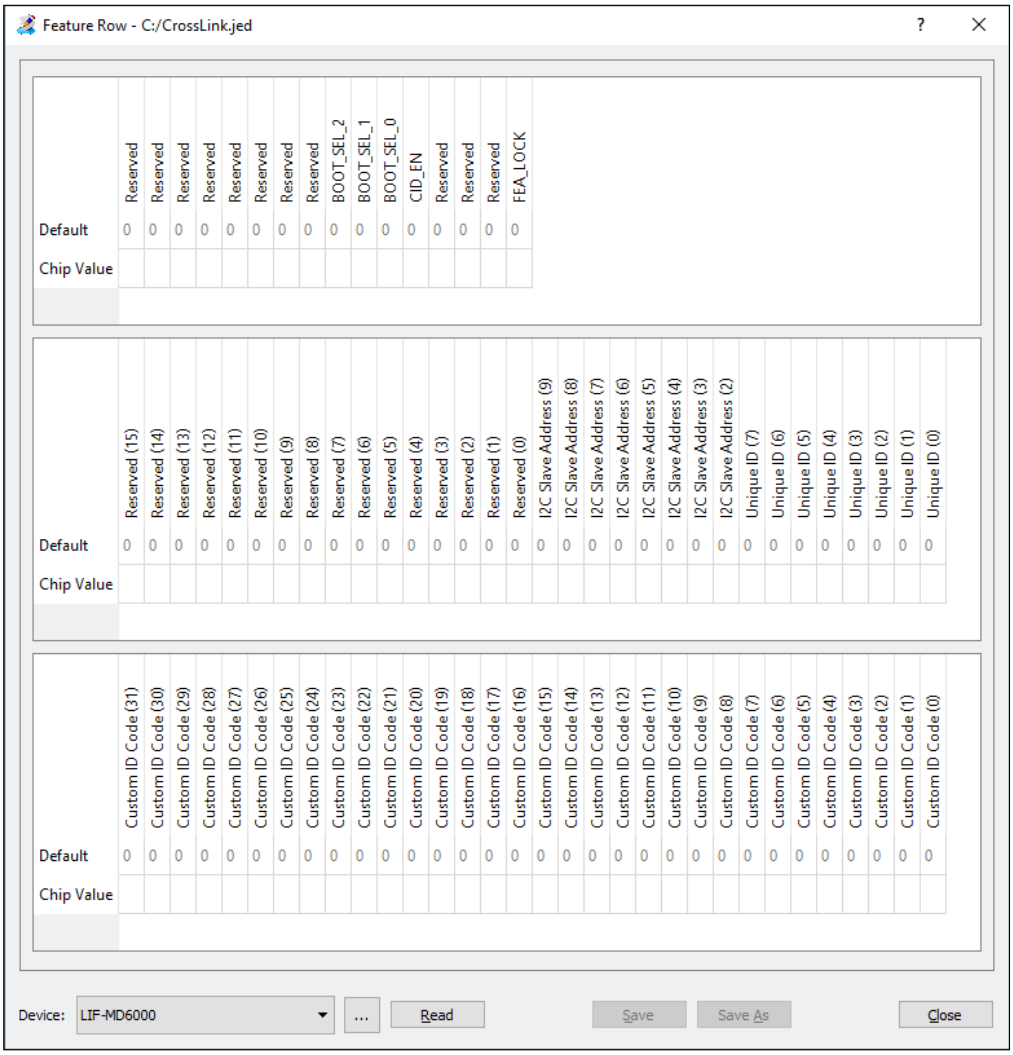

Figure 7.1. Programming File Utility – Feature Row Configuration Window

A full list of the functions controlled by the Feature Row and their default values for the CrossLink devices are shown in Table 7.1 and Table 7.2.

Table 7.1. Feature Row Definition

FEATURE ROW				
Bit	**Parameters**	**HW Default**	**SW Default**	**Description**
63	Reserved (15)	0	N/A	Reserved
62	Reserved (14)	0	N/A	Reserved
61	Reserved (13)	0	N/A	Reserved
60	Reserved (12)	0	N/A	Reserved
59	Reserved (11)	0	N/A	Reserved
58	Reserved (10)	0	N/A	Reserved
57	Reserved (9)	0	N/A	Reserved
56	Reserved (8)	0	N/A	Reserved
55	Reserved (7)	0	N/A	Reserved
54	Reserved (6)	0	N/A	Reserved
53	Reserved (5)	0	N/A	Reserved
52	Reserved (4)	0	N/A	Reserved
51	Reserved (3)	0	N/A	Reserved
50	Reserved (2)	0	N/A	Reserved
49	Reserved (1)	0	N/A	Reserved
48	Reserved (0)	0	N/A	Reserved
47	I^2C Slave Address (9)*)	h00	h00	The CrossLink I^2C configuration port supports 7-bit or 10-bit addresses. The lower two bits are always *00*. The I^2C Slave Address can be modified through the **I^2C Address** preference in the **Global Preferences** tab within **Spreadsheet View**.
46	I^2C Slave Address (8)*			
45	I^2C Slave Address (7)*			
44	I^2C Slave Address (6)*			
43	I^2C Slave Address (5)*			
42	I^2C Slave Address (4)*			
41	I^2C Slave Address (3)*			
40	I^2C Slave Address (2)*			
39	Unique ID (7)*	h00	h00	Unique ID is an 8-bit code that can be set by the customer. It can be modified through **Trace ID** preference in the **Global Preferences** tab within **Spreadsheet View**.
38	Unique ID (6)*			
37	Unique ID (5)*			
36	Unique ID (4)*			
35	Unique ID (3)*			
34	Unique ID (2)*			
33	Unique ID (1)*			
32	Unique ID (0)*			
31	Custom ID Code (31)	h00000000	h00000000	Used as the JTAG ID of the device when **MY_ASSP** preference is enabled. **MY_ASSP** can be enabled through **Global Preferences** tab within **Spreadsheet View**. Once **MY_ASSP** = ON, the **Custom IDCode** preference can also be modified within the **Global Preferences** tab.
30	Custom ID Code (30)			
29	Custom ID Code (29)			
28	Custom ID Code (28)			
27	Custom ID Code (27)			
26	Custom ID Code (26)			
25	Custom ID Code (25)			
24	Custom ID Code (24)			
23	Custom ID Code (23)			
22	Custom ID Code (22)			
21	Custom ID Code (21)			
20	Custom ID Code (20)			
19	Custom ID Code (19)			
18	Custom ID Code (18)			

FEATURE ROW				
17	Custom ID Code (17)			
16	Custom ID Code (16)			
15	Custom ID Code (15)			
14	Custom ID Code (14)			
13	Custom ID Code (13)			
12	Custom ID Code (12)			
11	Custom ID Code (11)			
10	Custom ID Code (10)			
9	Custom ID Code (9)			
8	Custom ID Code (8)			
7	Custom ID Code (7)			
6	Custom ID Code (6)			
5	Custom ID Code (5)			
4	Custom ID Code (4)			
3	Custom ID Code (3)			
2	Custom ID Code (2)			
1	Custom ID Code (1)			
0	Custom ID Code (0)			

*__Note:__ Can be modified in the Global Preference tab within the Spreadsheet View.

Table 7.2. Feature Bit Definition

FEATURE BITS				
Bit	**Parameters**	**HW Default**	**SW Default**	**Description**
15	Reserved	0	N/A	Reserved
14	Reserved	0	N/A	Reserved
13	Reserved	0	N/A	Reserved
12	Reserved	0	N/A	Reserved
11	Reserved	0	N/A	Reserved
10	Reserved	0	N/A	Reserved
9	Reserved	0	N/A	Reserved
8	Reserved	0	N/A	Reserved
7	BOOT_SEL_2*	b000	b010	**BOOT_SEL_[2:0]** 000 – NVCM-EXT 001 – EXT-NVCM 010 – NVCM 011 – EXT 100 – N/A 101 – EXT-EXT 110 – N/A 111 – N/A This can also be set through **BOOT_UP_SEQUENCE** in **Global Preference** tab within **Spreadsheet View**.
6	BOOT_SEL_1*			
5	BOOT_SEL_0*			
4	CID_EN*	0	0	This bit is set through **MY_ASSP** preference in **Global Preference** tab within **Spreadsheet View**. When enabled, **Custom IDCode**, in **Global Preferences** tab, can be modified.
3	Reserved	0	N/A	Reserved
2	Reserved	0	N/A	Reserved
1	Reserved	0	N/A	Reserved
0	FEA_LOCK*	0	0	This bit is set through **SECURITY_NVCM** preference in **Global Preferences** tab within **Spreadsheet View**. When the JEDEC file with this bit enabled is programmed to the CrossLink device, the Feature Row and Feature Bits become locked and can no longer be read or modified.

***Note:** Can be modified in the Global Preference tab within the Spreadsheet View.

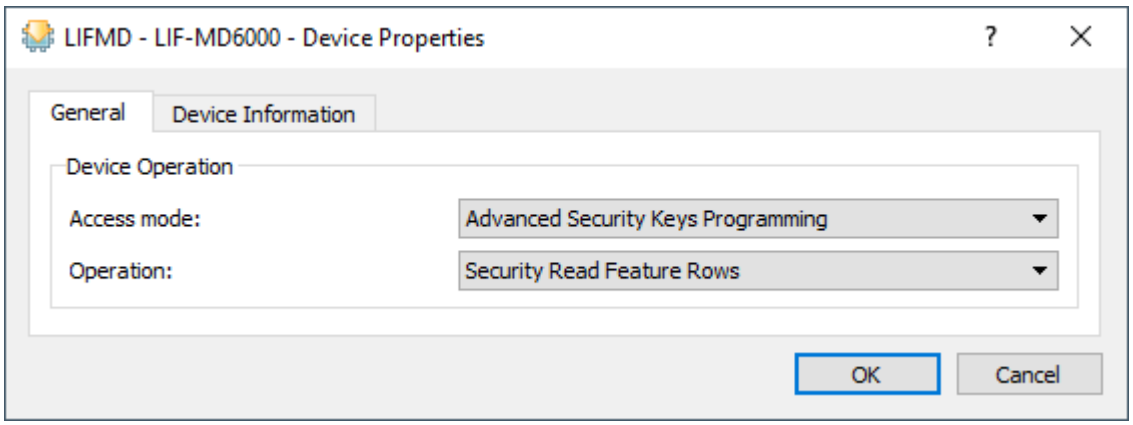

Figure 7.2. Diamond Programmer – Device Properties Window for Feature Row Read

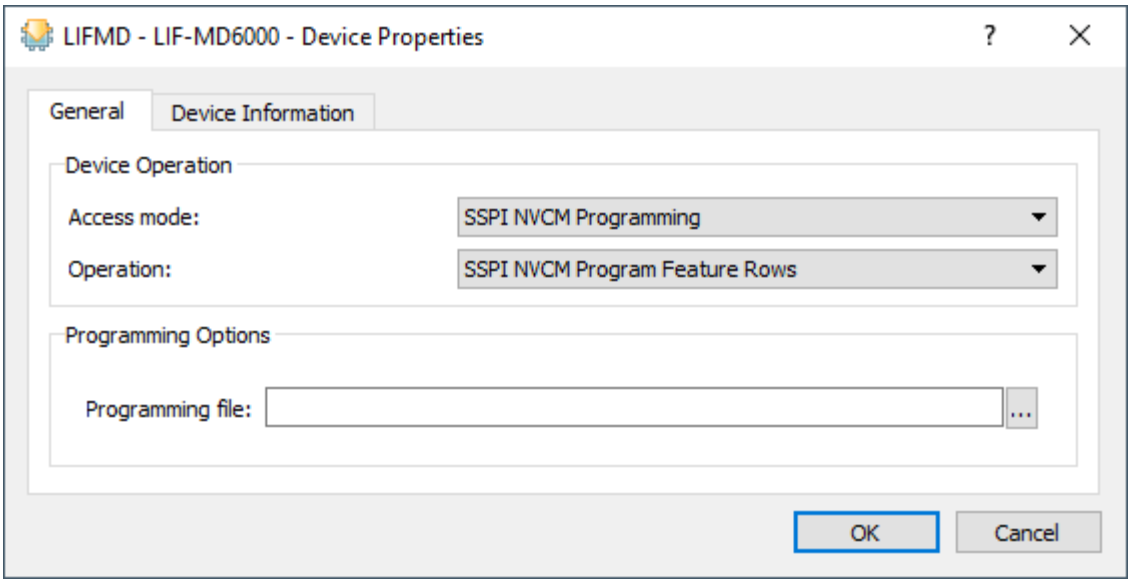

Figure 7.3. Diamond Programmer – Device Properties Window for Feature Row Programming

Appendix A. CrossLink SSPI SRAM Configuration Flow

		Shift In			Shift Out		
Steps	**Configuration Step**	**Signal**	**Clocks**	**Value**	**Signal**	**Value**	**Description**
1	Initialize	CRESETB		LOW			
		Wait 1 s					
		SN		LOW			
		SI	40	0xFF A4 C6 F4 8A			Shift in Activation Key
		SN		HIGH			
		CRESETB		HIGH			
2	Delay	Wait 10 ms					
3	Check IDCODE (Optional)	SN		LOW			
		SI	32	0xE0 00 00 00			Shift in IDCODE_PUB (0xE0) opcode
		SI	32	0x00 00 00 00	SO	DEVICE_ID	Shift out DEVICE_ID & compare to expected value
		SN		HIGH			
4	Enable Programming Mode	SN		LOW			
		SI	32	0xC6 00 00 00			Shift in ISC ENABLE (0xC6) instruction
		SN		HIGH			
5	Delay	Wait 200 ms					
6	Erase the device	SN		LOW			
		SI	32	0x0E 00 00 00			Shift in ISC ERASE (0x0E) instruction
		SN		HIGH			
7	Delay	Wait 200 ms					
8	Program Fuse Map	SN		LOW			
		SI	32	0x0E 00 00 00			Shift in LSC_INIT_ADDRESS (0x46) instruction
		SN		HIGH			
		SN		LOW			
		SI	32	0x7A 00 00 00			Shift in LSC_BISTREAM_BURST (0x7A) instruction
		SI		Full Bitstream			Shift in bitstream (.bit) generated in Diamond Software
		SN		HIGH			
9	Delay	Wait 10 ms					
10	Read the status bit	SN		LOW			
		SI	32	0x3C 00 00 00			Shift in LSC_READ_STATUS (0x3C) instruction
		SI	32	0x00 00 00 00	SO	Status Register Value	Expected Value from SO: 0x00000100 with the Mask: 0x00003100 [Bit-8 DONE = 1, Bit-12 Busy = 0, Bit-13 Fail = 0]
		SN		HIGH			
11	Exit Programming Mode	SN		LOW			
		SI	32	0x26 00 00 00	SO	Status Register Value	Shift in ISC DISABLE (0x26) instruction
		SN		HIGH			

		Shift In			Shift Out		
Steps	**Configuration Step**	**Signal**	**Clocks**	**Value**	**Signal**	**Value**	**Description**
		Wait 200 ms					
		SN		LOW			
		SI	32	0xFF FF FF FF			Shift in NO-OP (0xFF) instruction
		SN		HIGH			

Appendix B. CrossLink Slave I²C SRAM Configuration Flow

Steps	Signal	Configuration Step	Read/Write	Value	Description
1	CRESETB			LOW	
		Wait 1 s			
		Initialize	W	0xA4 0xC6 0xF4 0x8A	Shift in Activation Key
	CRESETB			HIGH	
		Wait 10 ms			
2		Check IDCODE (Optional)	W	0xE0 0x00 0x00 0x00	Shift in IDCODE (0xE0) opcode
			R	Byte 1 – Byte 2 – Byte 3 – Byte 4	Shift out DEVICE_ID & compare expected value
3		Enable SRAM Programming Mode	W	0xC6 0x00 0x00 0x00	Shift in ENABLE (0xC6) instruction
		Wait 1 ms			
4		Erase SRAM	W	0x0E 0x00 0x00 0x00	Shift in ERASE (0x0E) instruction
		Wait 5 s			
5		Read Status Register	W	0x3C 0x00 0x00 0x00	Shift in LSC_READ_STATUS (0x3C) instruction
			R	Byte 1 – Byte 2 – Byte 3 – Byte 4	Expected Value: 0x00000000 with the Mask: 0x00003000 [Bit-12 Busy = 0, Bit-13 Fail = 0]
6		Program SRAM	W	0x46 0x00 0x00 0x00	Shift in LSC_INIT_ADDRESS (0x46) instruction
			W	0x7A 0x00 0x00 0x00	Shift in LSC_BITSTREAM_BURST (0x7A) instruction
				Byte 1 – Byte 2 – … - Byte N	Shift in bitstream (.bit) generated by Diamond Software
		Wait 10 ms			
7		Verify USERCODE (Optional)	W	0xC0 0x00 0x00 0x00	Shift in USERCODE (0xC0) instruction
			R	Byte 1 – Byte 2 – Byte 3 – Byte 4	Read USECODE
8		Read Status Register	W	0x3C 0x00 0x00 0x00	Shift in LSC_READ_STATUS (0x3C) instruction
			R	Byte 1 – Byte 2 – Byte 3 –Byte 4	Expected Value from 0x00000100 with the Mask: 0x00003100 [Bit-8 DONE = 1, Bit-12 Busy = 0, Bit-13 Fail = 0]
9		Exit Programming Mode	W	0x26 0x00 0x00 0x00	Shift in DISABLE (0x26) instruction

Appendix C. CrossLink Status Register Map

The User Status Register (USR) is used to provide the information of the device during or after the configuration process to the user through the LSC_READ_STATUS command. The default value of the Status Register is '0' and it is reset by powering up, PROGRAMN pin toggle low, or the REFRESH command execution. All status register bits are non-sticky unless otherwise specified.

Definition	Bit	Note
Fail Flag	13	Status bit indicates that the last command execution failed
Busy Flag	12	Status bit indicates that the configuration logic is currently busy executing a previous command
Done	8	Status bit indicates that the device has completed configuration and the DONE bit is set

References

For more information, refer to the following documents:

- CrossLink Family Data Sheet (FPGA-DS-02007)
- CrossLink High-Speed I/O Interface (FPGA-TN-02012)
- CrossLink Hardware Checklist (FPGA-TN-02013)
- CrossLink sysCLOCK PLL/DLL Design and Usage Guide (FPGA-TN02015)
- CrossLink sysI/O Usage Guide (FPGA-TN-02016)
- CrossLink Memory Usage Guide (FPGA-TN-02017)
- Power Management and Calculation for CrossLink Devices (FPGA-TN-02018)
- CrossLink I^2C Hardened IP Usage Guide (FPGA-TN02019)
- Advanced CrossLink I^2C Hardened IP Reference Guide (FPGA-TN-02020)
- Minimizing System Interruption During Configuration Using TransFR Technology (TN1087)
- Programming Tools User Guide

Technical Support Assistance

Submit a technical support case through www.latticesemi.com/techsupport.

Revision History

Revision 1.5, July 2021

Section	Change Summary
Appendix A. CrossLink SSPI SRAM Configuration Flow	Revised Appendix A Step 8 to LSC_INIT_ADDRESS (0x46).
Appendix C. Crosslink Status Register Map	Appended this section Appendix C.

Revision 1.4, March 2021

Section	Change Summary
Software Selectable Options	Added *This only applies to User Mode and does not affect the programming of CrossLink during configuration* information in Configuration Mode and Port Options section.
Feature Row	• Updated section content to add reference to Diamond Programmer Feature Row Read and Feature Row Programming. • Added Figure 7.2 and Figure 7.3.
Appendix A. CrossLink SSPI SRAM Configuration Flow	Added this section.
Appendix B. CrossLink Slave I^2C SRAM Configuration Flow	Added this section.

Revision 1.3, December 2020

Section	Change Summary
All	Updated format of last page.
Disclaimers	Added this section.
CrossLink Features	Updated content.
Definition of Terms	• Added definition for Feature Row HW Default Mode. • Updated table content.
Configuration Process Flow	• Updated Figure 4.1. • Updated content of Power-up Sequence, Configuration Ports Default Behavior and Arbitration, Configuration, Wake-up, User Mode, Clearing the Configuration Memory and Re-initialization, Configuration Modes of CrossLink, and sysCONFIG Pins. • Updated Table 4.1, Table 4.3, Table 4.5, and Table 4.6.
Configuration Modes	• Updated content of Master SPI Configuration Mode, Dual Boot Configuration Mode, Slave SPI Mode, and I2C Configuration Mode. • Updated Table 5.2, Table 5.3, Table 5.5, and Table 5.7.
Software Selectable Options	• Updated content of Configuration Mode and Port Options, Bitstream Generation Options, and Security Options. • Updated Table 6.1 and Table 6.2.
Feature Row	• Changed section name from Device Wake-up Sequence to *Feature Row*. • Updated content.
Revision History	Updated format.

Revision 1.2, December 2017

Section	Change Summary
Configuration Process and Flow	Updated this section. Removed references to Table 4.1.
Power-up Sequence	Updated this section • Added information on upstream sources • Changed $V_{CCAUX25VPP}$ to V_{CCAUX}.
Initialization	Updated this section

Configuration Process and Flow	• Updated the Configuration Ports Default Behavior and Arbitration section
	  • Changed "toggle LOW or REFRESH" to "toggle from LOW to HIGH or REFRESH"
	  • Added information on the Activation Key
	• Updated the Configuration section. Added information on the Activation Key
	• Updated the Clearing the Configuration Memory and Re-initialization section. Added content to the third method of clearing the internal configuration memory
	• Updated the Self-Download Port Pins section. Revised the first and last paragraphs.
	• Updated the Master and Slave SPI Configuration Port Pins section. Updated the CRESET_B direction and description in Table 4.5. Master SPI Configuration Port Pins and Table 4.6. Slave SPI Configuration Port Pins

Configuration Modes	• Updated the last paragraph in SDM Mode section
	• Updated the Master SPI Configuration Mode section
	  • Updated configuration data flow in the fourth paragraph
	  • Updated SPI Flash configuration information in the last paragraph
	• Added information on the Activation Key to the Slave SPI Mode section
	• Added information on the Activation Key to the I2C Configuration Mode section

Device Wake-up Sequence	Updated introductory paragraph of this section

Revision 1.1, February 2017

Section	Change Summary
Configuration Process and Flow	• Updated the Configuration Ports Default Behavior and Arbitration section with default behavior • Updated the Configuration section with two cases • Added Note 2 to Table 4.4. Default State in Diamond for each Port • Changed CCLK to "MCK or SPI_SCK", MCK to CSN, and general purpose I/O to SPI_SS in the Master and Slave SPI Configuration Port Pins section
Configuration Modes	• Changed MCLK to MCK in Master SPI Configuration Mode section • Added reference to the Programming Tools User Guide

Revision 1.0, August 2016

Section	Change Summary
All	Updated document numbers, the previous document number was TN1303.

Revision 1.0, May 2016

Section	Change Summary
All	First preliminary release.

www.latticesemi.com